音声スタータ：有声休止による発話開始の指定が可能な音声入力インタフェース

後藤真孝[†] 北山広治[††,☆]
伊藤克亘[†††,☆☆] 小林哲則[††]

本論文では，ユーザが有声休止（母音の引き延ばし）によって言い淀んだ後に音声入力することで，雑音環境下での発話区間検出を容易にする「音声スタータ」という音声インタフェース機能を提案する．通常の音声認識システムでは，入力音響信号から発話区間を検出した後に，その区間に対して音声認識結果を得る．しかし非定常な雑音環境下では，頑健に発話区間を検出することが困難なため，音声認識誤りを生じることが多かった．音声スタータでは，ユーザが「えー」や「あのー」のように有声休止を発話の先頭（発話区間の始端）で故意に発声することで，システムに音声認識してほしい発話を明示的に指定することを可能にする．有声休止はパワーの大きい母音が持続することから，雑音環境下でも頑健に検出でき，発話区間検出の精度を向上させることができる．さらに，音声スタータではマイク以外のデバイスが不要でハンズフリーな音声認識を実現でき，日常会話でも言い淀んでから話し始めることがよくあるためにユーザの負担も少ないという利点がある．実際に 7 種類の雑音環境下で音声認識実験をしたところ，特に SNR 10 dB において従来の他の発話区間検出手法を用いた場合よりも，音声スタータを用いた場合の方が検出性能が高かった．

Speech Starter: Speech Input Interface Capable of Endpoint Detection by Using Filled Pauses

Masataka Goto,[†] Koji Kitayama,[††,☆] Katunobu Itou[†††,☆☆] and Tetsunori Kobayashi[††]

This paper describes a speech interface function, called *Speech Starter*, which enables noise-robust endpoint (utterance) detection by having a user utter a filled pause (a vowel-lengthening hesitation) at the beginning of each utterance. Most current speech recognizers first detect a utterance with its endpoints and then recognize the detected utterance. When speech recognizers are used in a noisy environment, a typical recognition error is caused by incorrect endpoints because their automatic detection is likely to be disturbed by non-stationary noise. Speech Starter enables a user to specify the beginning of each utterance with an intentional filled pause (e.g., "er..."), which is used as a trigger to start speech-recognition processes. Because a filled pause contains a lengthened vowel with high power and can be detected robustly in a noisy environment, practical robust endpoint detection is achieved. Speech Starter also offers the advantage of providing a hands-free speech interface with a microphone only, and it is user-friendly because a speaker tends to utter filled pauses at the beginning of utterances when hesitating in human-human communication. Experimental results with seven different noisy environments show that Speech Starter achieved the higher detection performance than conventional endpoint detection methods, especially at the SNR of 10 dB.

† 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology（AIST）
†† 早稲田大学
Waseda University
††† 名古屋大学
Nagoya University
☆ 現在，株式会社東芝
Presently with Toshiba Corporation
☆☆ 現在，法政大学
Presently with Hosei University

1. はじめに

本論文では，非定常な雑音に頑健な音声認識システムの構築を目的として，有声休止による発話開始の指定手法について検討する．一般的な音声認識システムは，入力音声信号から発話区間（本論文では音声認識処理を行う区間を意味する）を同定後，その区間に対して音声認識する．そのため，発話区間検出精度が音声認識システム全体の性能に大きく影響する．典型的な発話区間の検出方法として，零交差数と短時間エネ

ルギーの 2 つの音響特徴量に基づいて発話区間を検出する方法[1] が古くから使われており，静かな室内等の雑音の少ない環境下では適切に発話区間検出が可能である．しかし，実環境で入力される音声の場合，音声以外の雑音の影響によって，検出精度が低下するという問題が生じていた．また，咳払いや息等の，ユーザがシステムに入力するつもりのない音が，発話区間と誤検出されて音声認識誤りを起こすという問題もあった．そのため，ユーザは咳払い等をせずに誤りなく音声入力しなければならず，音声認識システムの操作性を損なっていた．

上記の問題に対処する代表的な解決法として，マイク以外のデバイスを使用する方法があげられる．たとえば，ユーザが音声入力のために話している間，ボタンを押し続けてシステムに発話区間を指示する方法がある．しかし，実際にはユーザがボタンを押すタイミングが早過ぎたり遅過ぎたりすることがあるため，特に雑音環境下では発話区間に雑音が含まれてしまい，頑健性を損なうことがあった．また，ボタン操作が前提なために，他のデバイスを使わないマイク入力だけのシステムは実現できなかった．ほかにもマイク以外のデバイスを用いる方法としては，カメラでとらえた話者の顔の動きに基づいて発話区間を切り出したり不要な発話を棄却したりする方法[2),3)] が研究されている．しかし，これらの方法はカメラが利用できない環境には適用できず，ユーザの行動範囲も制限されていた．一方，そうした制約のない，マイクだけを用いた解決法としては，発話区間検出に用いる音響特徴量の使用方法を改善する研究[4)~6)] が提案されているが，非定常な雑音に対しては，まだ性能向上の余地がある．また別の解決法として，発話区間を明示的に検出せずに，連続して音声認識をし続ける手法[7),8)] が提案されている．これらの方法は，発話を検出し損なう可能性は少ないが，ユーザが入力するつもりのない音（咳払いや息等）や背景雑音を誤って認識対象としてしまう可能性があり，これらの音を何らかの方法で適切に棄却しなければならなかった．

そこで我々は，上記の問題を解決する新しい音声インタフェース機能「音声スタータ」を提案する．人間は話し始めるときに，しばしば有声休止（母音の引き延ばし）を含む「えー」や「あのー」といったつなぎ語をいって言い淀むことがある．音声スタータでは，この音声の非言語情報の 1 つである有声休止に着目し，音声認識を開始するトリガとして活用する（音声認識器は，有声休止から認識処理を開始する）．有声休止は，自然対話中で頑健に検出できることが報告されている[9)]．有声休止はパワーが大きく安定した母音をともなうことから雑音環境下でも検出可能であり，これを発話区間の始端と見なせば，頑健な発話区間検出が期待できる．そこで音声スタータでは，ユーザが発話開始時につねに有声休止を発声することをルール化する．こうすることで，認識してほしい発話区間を，その先頭で故意に言い淀むことによってユーザ自身が明示的に指示できるという利点が得られる．

以下，2 章において，提案する音声スタータの利点を議論し，具体的な実現方法を説明する．次に，3 章で音声スタータを組み込んだ音声認識システムの実装について述べる．そして，4 章で雑音に対する頑健性の評価実験の結果を述べ，音声スタータの有効性を示す．最後に，5 章でまとめと今後の課題を述べる．

2. 音声スタータ

音声スタータは，ユーザが音声だけで音声認識器の認識開始点（発話区間の始端）を指示できる音声インタフェース機能である．ユーザが，認識してほしい各発話の先頭で必ず有声休止を発声して言い淀むことで，音声認識開始点をユーザ自身が制御可能となる．発話の先頭で言い淀む際には，有声休止を末尾に含む様々なつなぎ語を用いてもよいが，認識対象の語句との間に長い無音が挿入されないように，連続して発声するものとする（ただし，連続して話しているように聞こえる程度の短い無音は挿入されてもよい）．たとえば，ユーザが「藤井フミヤ」という人名を音声入力したい場合，「えー，藤井フミヤ」や「あのー，藤井フミヤ」のように，故意に言い淀んだ直後に認識してほしい語句を発声すればよい．このように，音声スタータでは，非言語情報（有声休止）を用いてユーザが能動的に発話区間の始端を伝えることができる点が，従来なかった新たなインタフェース機能となっている．

音声スタータにより，以下の 3 つの利点が得られる．

(1) 雑音に頑健な発話区間検出
母音のパワーは音声信号中で比較的大きく，それが引き延ばされた有声休止も同様に大きいパワーを持つことから，雑音環境下で頑健に検出しやすい．雑音環境下で音声認識する際には，本来の発話の直前に混入した雑音から認識処理を開始して誤認識してしまう問題が生じることがあるが，音声スタータではつねに安定した母音の途中から認識処理を開始するため，そうした問題を回避できる．また，有声休止が検出されるまでは認識処理を開始しないので，非定常な突発的雑音を発話区間と誤検出することが

ない．
なお，音声スタータでは，様々な発話の中から認識対象となる発話だけを検出（スポッティング）する用途は想定しておらず，ユーザは認識対象の発話以外は発声しないという条件下で，雑音環境下での発話区間の検出を可能にする．

(2) ユーザの負担が少ない操作
音声スタータは特別な訓練をせずに使うことができ，ユーザの負担も少ない．音声スタータでは，ユーザは故意に有声休止を発声してから話し始めなければならないが，日常会話でも言い淀んでから話し始めることがよくあるため，そのような話し方には慣れている．

(3) マイク以外のデバイスが不要
音声スタータは音声のみで使用可能なため，ボタンやカメラ等のマイク以外のデバイスを必要としない．そのため，音声スタータを組み込んだアプリケーションシステムはコンパクトに実現することができる．また，ユーザの行動範囲も制限されることがない．

このような音声スタータの機能を実現するには，まず，リアルタイムに有声休止を自動検出する必要がある．次に，検出結果に基づいて音声認識器の認識開始点を決定し，実際の認識処理を始めなければならない．最後に，音声認識器の認識終了点を，音声入力中の刻々と変化する認識結果に基づいて決定する必要がある．以下では，これらの具体的な方法を順に説明する．

2.1 有声休止の検出

言語的な制約をいっさい用いずに有声休止を検出するために，文献 9) のリアルタイム有声休止検出手法を用いる．この手法は，有声休止（母音の引き延ばし）が持つ 2 つの音響的特徴（基本周波数の変動が小さい，スペクトル包絡の変形が小さい）をボトムアップな信号処理によってリアルタイムに検出する．つなぎ語中の任意の有声休止の始端と終端を，言語非依存に検出できるという特長を持っている．具体的な検出手順を以下に述べる．

2.1.1 基本周波数の推定

雑音環境下で頑健に機能するように，LPC 等の単一音源を前提とした分析は行わず，入力信号中で最も優勢な（パワーの大きい）高調波構造の基本周波数を，音声の基本周波数（声の高さ）として抽出する．そこで，コムフィルタの考え方に基づいて，時刻 t において☆周波数 F が基本周波数となる可能性 $P_{F0}(F,t) = \int_{-\infty}^{\infty} p(x;F)\ \Psi_p(x,t)\ dx$ を評価する．$p(x;F)$ は基本周波数が F の高調波成分を通過させるフィルタ関数，$\Psi_p(x,t)$ は周波数成分のパワー分布関数とする．$P_{F0}(F,t)$ は各高調波構造が相対的にどれくらい優勢かを表すため，話者の音声の基本周波数 $F_{F0}(t)$ は，$F_{F0}(t) = \mathrm{argmax}_F\ P_{F0}(F,t)$ で求まる．

2.1.2 スペクトル包絡の推定

雑音環境下で頑健に包絡を推定するために，$F_{F0}(t)$ の高調波構造上にある局所的な情報だけを利用する．まず，$F_{F0}(t)$ の整数倍の周波数を中心とするガウス分布で重み付けしながら，その近傍の最大パワーを検出することで，各高調波成分のパワーを求める．次に，隣接する成分のパワーの間を直線補間してスペクトル包絡を求める．有声休止を検出するためには，包絡の大局的な変形をとらえた方がよいため，直線補間した包絡を粗い周波数分解能でリサンプリングし，低い方から n（$1 \le n \le \mathrm{N_{max}}$）点目の周波数におけるスペクトル包絡 $Env(n,t)$ を求める（$\mathrm{N_{max}}$ は文献 9) と同様に 15 とした）．最後に，呼気による AM 変調の影響を除去するために $Env(n,t)$ を正規化する．

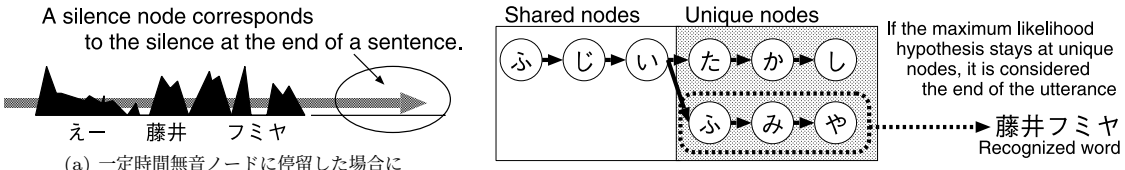
2.1.3 有声休止区間の決定

有声休止を検出するための 2 つの特徴量として，基本周波数の変動量 $A_f(t)$ とスペクトル包絡の変形量 $A_s(t)$ を求める．これらは，$F_{F0}(t)$ と $Env(n,t)$ の過去一定期間の対数スケール上での変化を，最小自乗法で直線近似した直線の傾き $b_f(n)$，$b_s(n)$ と近似誤差 $err_f(n)$，$err_s(n)$ を用いて，$A_f(t) = |b_{F0}|$，$A_s(t) = \left(\frac{1}{\mathrm{N_{max}}}\sum_{n=1}^{\mathrm{N_{max}}} b_s(n)^2\right)\left(\frac{1}{\mathrm{N_{max}}}\sum_{n=1}^{\mathrm{N_{max}}} err_s(n)^2\right)$ のように定義される．そして，有声休止らしさ（有声休止と判定する信頼度）$P_{fp}(t)$ を，$A_f(t)$，$A_s(t)$ の短時間平均 $S_f(t)$，$S_s(t)$ に基づいて，$P_{fp}(t) = \exp\left(-\frac{(\mathrm{R}\ S_f(t) + (1-\mathrm{R})\ S_s(t))^2}{\mathrm{W}^2}\right)$ と定義する．R は特徴に対する重み付けを決める定数，W は変動･変形の考慮範囲を決める定数である．有声休止検出率が高くなるように，文献 9) の実験結果に基づいて，$\mathrm{R} = 0.011$，$\mathrm{W} = 0.39$ と設定した．最終的に，$P_{fp}(t)$ が十分高い値をとり続けたときに有声休止の始端と判定し，再び小さくなったら終端と判定する．

2.2 発話始端（音声認識開始点）の決定

検出した有声休止区間の終端に基づいて，発話区間の始端を決定する様子を**図 1** に示す．有声休止区間の途中に発話始端が位置するようにし，有声休止の末尾も含めて音声認識する．これは，有声休止区間の終端付近に音素遷移の過渡的な現象が現れることがあり，終端後から音声認識すると誤認識を招くことがあるか

☆ 現在の実装では，16 kHz / 16 bit で A/D 変換し，フレームシフト 10 msec（160 点）をすべての処理の時間単位とする．



(a) 一定時間無音ノードに停留した場合に音声認識を終了する

(b) 一定時間固有ノードに停留した場合に音声認識を終了する

図 2 発話終端（音声認識終了点）の決定
Fig. 2 Determining the end of an utterance.

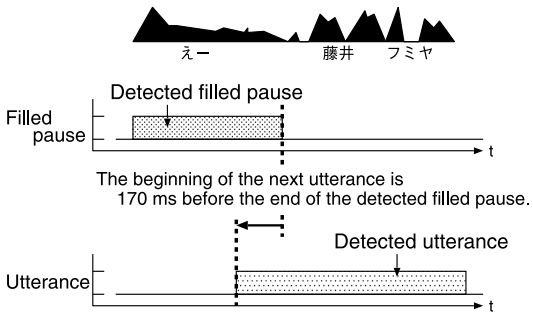


図 1 発話始端（音声認識開始点）の決定
Fig. 1 Determining the beginning of an utterance.

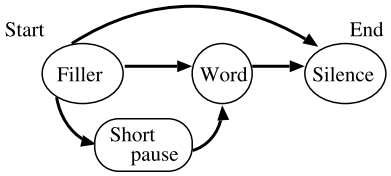


図 3 音声スタータの文法
Fig. 3 Grammar for Speech Starter.

らである．有声休止区間の途中ならば安定した母音であることが分かっているので，そこを発話始端とすることで，より適切に母音から認識対象の発話部分へと続く区間を音声認識できる．ただし，有声休止区間の始端を発話始端としてしまうと，音声認識対象となる有声休止の区間が長くなって誤認識を招く可能性が増えるため，必要以上に長くしない方がよい．予備実験の結果では，50 ms から 130 ms の範囲では長くするほど単調に性能が向上したが，130 ms から 200 ms の範囲では性能が飽和してほとんど変化しなかった．これらを考慮して，現在の実装では，有声休止区間の終端から 170 ms 手前の時点を発話始端としている．

2.3 発話終端（音声認識終了点）の決定

発話区間の終端は，パワー等で判断するのではなく，音声認識器の出力をもとに，内藤らの方法[10] と井ノ上らの方法[11] に基づいて決定する．具体的には，各フレーム（10 ms）において音声認識途中の最尤仮説を調べ，以下にあげる 2 つのノードのいずれかに一定時間以上（現在の実装では 200 ms 以上）停留している場合，そのフレームを発話終端と判定する．

(1) 無音ノードに停留している場合
最尤仮説が，文末の無音に対応した無音ノードに停留している場合，発話終端と見なす[10]．**図 2** (a) に例示したように，単純に，最尤仮説が連続して無音の状態であれば，文の途中でなく発話が終わったと判断できる．ただし，今回は単語認識を対象とし，**図 3** に示すネットワーク文法を使用して音声認識する．この文法では，有声休止を含むつなぎ語（Filler）から認識対象となる単語（Word）に直接遷移するか，短い無音（Short pause）を経て遷移し，最終的に無音（Silence）に到達する．ただし，つなぎ語としては有声休止以降を登録する．たとえば，「えー」は「えー」を登録するが，「あのー」は末尾の母音の「おー」の部分だけを登録する．

(2) 固有ノードに停留している場合
最尤仮説が，木構造辞書中で他の単語と共有されていない固有ノード（1 単語のみに対応し，他の単語の可能性がなくなったノード）に停留している場合，発話終端と見なす[11]．たとえば，図 2 (b) のように，「藤井隆」と「藤井フミヤ」の 2 単語を持つ木構造辞書を探索中とする．図 2 (b) の破線で囲まれている「ふ」「み」「や」のノードは，「藤井フミヤ」のみの独立した固有ノードで，「藤井」の部分のように他の単語とは共有されていない．そこで，最尤仮説がそれらの固有ノードに達した後に停留し続けていれば，「藤井フミヤ」であると十分判断できるとして，発話の終端とする．

3. 実　　装

図 4 に，音声スタータを用いた音声認識インタフェースのプロトタイプシステムの各構成要素（プロセス）と，全体の処理の流れを示す．プロセスは図中の囲み字で示されており，ネットワーク（LAN）上の複数の計算機で分散して実行することができる．プロセス間

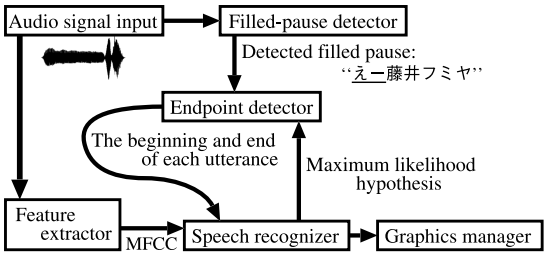


図 4 システム構成
Fig. 4 System architecture.

の通信には，音声言語情報をネットワーク上で効率良く共有することを可能にするネットワークプロトコル RVCP（Remote Voice Control Protocol）[12] を用いた．音声認識部は，発話終端を決定するために，毎フレームの最尤仮説を発話区間検出部に送信できる必要がある．そこで，CSRC の日本語ディクテーション基本ソフトウェア（julian 3.3beta）[13),14)] を RVCP に対応させ，そうした最尤仮説の送信が可能なように拡張して用いた．

処理の流れについて説明する．まず，マイク等から音声入力部（Audio signal input）に入力された音響信号は，ネットワーク上にパケットとして送信される．有声休止検出部（Filled-pause detector）と特徴量抽出部（Feature extractor）がそのパケットを同時に受信し，両者で並行して処理する．次に，発話区間検出部（Endpoint detector）は，有声休止区間の終端情報を有声休止検出部から受け取って発話始端を決定し，音声認識部（Speech recognizer）にその情報を送信する．音声認識部は特徴量抽出部から MFCC（mel-frequency cepstral coefficients）パラメータを受け取り，検出された発話始端から認識処理を開始する．そして，最尤仮説を毎フレームごとに発話区間検出部に送る．発話区間検出部は受け取った最尤仮説をもとに，発話終端の 2 つの条件のいずれかを満たしているかどうか判定して発話終端を決定し，その情報を音声認識部に送信する☆．最後に，音声認識部が発話終端時点の認識結果を認識結果表示部（Graphics manager）へ送信する．

☆ 仮に音声認識部が現在の発話を認識するのに多少時間がかかり，次の発話の有声休止が発声され始めたとしても，別のプロセスである有声休止検出部が検出して，その情報が RVCP のパケットとして送受信される．このように複数のプロセスで分散処理をしているため，現在の発話の処理が終了した後に，次の発話の有声休止区間の終端情報に基づいて，音声認識部が適切に次の発話の処理を開始することが可能である．

4. 性能評価

音声スタータが雑音環境下で頑健に機能することを確認するために，多様な雑音を様々な SNR で混合した音声データを用いて，以下の 4 つの発話区間検出手法を比較評価する実験をした．

(1) 音声スタータ
(2) 零交差数と短時間エネルギーに基づいて発話区間を検出する方法[1)]
(3) CSRC の日本語ディクテーション基本ソフトウェア julian [13),14)] に実装されている，発話区間を検出せずに音声認識を行うために短い無音（ショートポーズ）でセグメンテーションを行う方法[8)]
(4) 発話区間の先頭で，つねにキーワードを発声することをルールとし，キーワードに基づいて発話区間の始端を決定する方法

キーワードを用いる (4) の方法は，音声スタータの有声休止の代わりにキーワードを用いた場合を比較検討するために評価した☆☆．キーワードの選択は任意であるが，計算機に話しかけるときに用いられても不自然でなく，かつ，発声しやすい単語の中から，キーワードとして「もしもし」と「コンピュータ」を採用した．たとえば，「もしもし，藤井フミヤ」のように発声することで，音声スタータと同じように発話区間の始端をユーザが明示的に指示できる．この場合，音声認識システムはキーワードを検出してから認識処理を開始する．文献 15) を参考にしたキーワードの検出（ワードスポッティング）と認識処理の手順を以下に示す．

(i) 上記の (3) の方法[8)] で音声認識する．
(ii) キーワードと認識された区間に対して，文法の拘束をなくした状態で（すべての音素間の遷移を許可した文法を用いて），もう一度認識処理をして尤度を求める．
(iii) 文法の拘束をなくした状態の結果を対立候補と考え，その尤度を，元のキーワードを認識した際の尤度と比較する．
(iv) 両者の尤度が近ければ，キーワードが発声されたと判定し，それに続く認識結果を受理する．そうでなければ棄却する．

☆☆ 認識対象の発話の直前につねに特定のキーワードを発声することをユーザに義務付けるアプローチは，古くから存在する．たとえば，「コンピュータ」のような一般的な語や，「Casper」，「Maxwell」のようなシステム名をキーワードとしてスポッティングし，認識対象となる発話を特定していた．

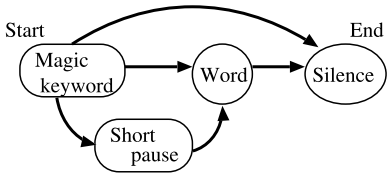


図 5 キーワードを用いる方法の場合の文法
Fig. 5 Grammar for the keyword-based method.

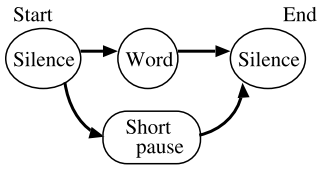


図 6 他の方法の場合の文法
Fig. 6 Grammar for the other methods.

認識対象（音声入力対象）の単語辞書としては，日本のポピュラー音楽のヒットチャート（2000 年度のすべての週間ランキングのシングル上位 20 曲）から作成した曲名（342 語）とアーティスト名（179 語）の単語辞書（計 521 語）[12] を用いた．各曲名とアーティスト名を音声認識器の単語辞書上の 1 単語として登録し，音声スタータの場合には，図 3 に示すネットワーク文法を使用して音声認識した．キーワードを用いる方法の場合の文法を**図 5** に，それ以外の方法の場合の文法を**図 6** に示す．図 3 では，つなぎ語の有声休止以降の部分（Filler）として，「あー」「いー」「うー」「えー」「おー」「んー」を登録した．図 5 では，キーワード（Magic keyword）として「もしもし」「コンピュータ」を登録した．図 6 では，無音（Silence）が最初のノードとなる．

4.1 実験条件

図 7 に，4 手法の評価に用いた音声データの作成方法を示す．共通の発話を用いて発話区間検出の性能を比較するために，個々の発話を個別に収録し，それらを連結した後に雑音を混合することで評価用データを作成した．

まず 20 代の男性話者 10 人から，1 人あたり以下の 70 発話を収録し，発話の前後に無音をあまり含まないように計 700 発話を切り出した．

- 単語辞書上の単語（アーティスト名）50 語を発声した 50 発話
- 有声休止を含むつなぎ語「えー」を 6 回発声した 6 発話
- 有声休止を含むつなぎ語「あのー」を 4 回発声した 4 発話
- キーワード「もしもし」を 5 回発声した 5 発話

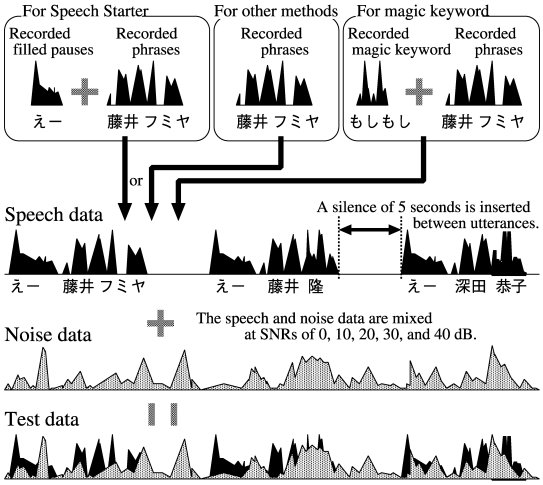


図 7 評価用データの作成方法
Fig. 7 Method of preparing the test data.

- キーワード「コンピュータ」を 5 回発声した 5 発話

文献 16) では，日本語の自由発話におけるつなぎ語（間投詞）の種類と出現頻度を調査しており，その結果によれば，最も多く出現するのは「えー」で，その次に「あのー」が多く出現することが分かる．そこで，この調査結果の両者の出現頻度の比（約 6:4）に合わせて，「えー」を 6 発話，「あのー」を 4 発話，収録することとした．キーワードを用いる方法では，「もしもし」と「コンピュータ」の 2 種類を 5 発話ずつ収録することとした．

次に，音声スタータの評価のために，収録した有声休止と単語を連結して 1 つの単位となる発話（単位発話）を作成した．キーワードを用いる方法のためにも同様に，キーワードと単語を連結して 1 つの単位発話を作成した．上記の連結の際には，息継ぎをせずに連続して発声したように聞こえるよう，長い無音が挿入されないように注意深く連結した（ただし，数十 ms の短い無音が挿入されないと，連結した前後で不自然に聞こえるため，自然に聞こえるように試行錯誤して調節した）．他の方法の評価データには，単語をそのまま単位発話として用いればよい．そして，発話区間検出の性能を評価できるように，これらの単位発話の間に 5 秒間の無音を挿入しながら連結した．こうして，50 個の単位発話を連結した音声データを 3 種類作成した．このように，単語辞書上の単語の発声自体は，比較する 4 つの発話区間検出手法で共通であり，発話区間検出の性能に焦点を当てた評価が可能となる．

それから，作成した音声データに，下記の 7 種類の実環境雑音[17] を 5 種類の SNR（0，10，20，30，

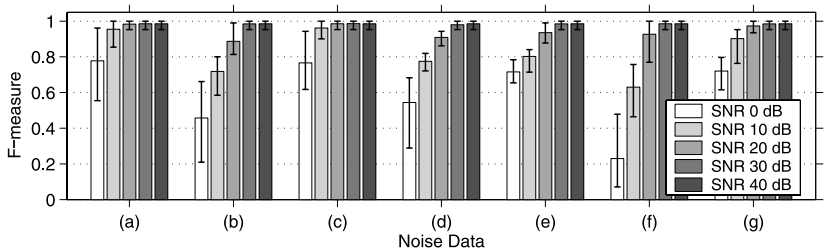


図 8 7 種類（(a)〜(g)）の各実環境雑音下での有声休止検出率：各実環境雑音における 5 つの棒グラフが 5 種類（0，10，20，30，40 dB）の SNR での 10 人の発声に対する検出率の平均値を表し，それに付随する高低線が 10 人の発声の中での最大値と最小値を表す
Fig. 8 Filled-pause detection rates with seven different noisy environments.

40 dB）で混合した．SNR は，単位発話 50 個の全区間の平均エネルギーと，それと同区間の雑音の平均エネルギーから計算した．

(a) 走行自動車内［1,500 cc クラス］
低域にエネルギーが集中している．

(b) 展示会場［ブース内］
ときどき人の話し声が聞こえる．

(c) 展示会場［通路］
ときどき人の話し声が聞こえる．

(d) 交差点
車の通行音が聞こえる．

(e) 列車［在来線］
断続的に列車の激しい走行音がする．

(f) 計算機室［ワークステーション］
定常的なファンの音が聞こえる．

(g) エレベーターホール［百貨店］
人の雑踏や話し声が聞こえる．

本実験では，MFCC 12 次元 + ΔMFCC 12 次元 + Δpower 1 次元の計 25 次元の音声特徴量を用いた．音響モデルは，ASJ-JNAS，ASJ-PB の男性話者 133 人分（計 20,414 文）[18] から学習し，混合数 16，状態数 2,000 のトライフォンモデルとした．

4.2 評価方法

評価用データの各発話区間の出現位置とその発話の正解単語を記述した正解単語ラベルと，システムの出力（各発話区間とその認識結果）を比較する．両者が一致している度合いを，再現率（recall rate）R，適合率（precision rate）P，および両者を統合した F 値（F-measure）[19] の観点から評価した．以下に定義を示す．

$$R = \frac{\text{正しく認識した発話区間の数}}{\text{正解の発話区間の数（50 個）}} \quad (1)$$

$$P = \frac{\text{正しく認識した発話区間の数}}{\text{検出された発話区間の数}} \quad (2)$$

$$\text{F 値} = \frac{(\beta^2+1)PR}{\beta^2 P + R} \quad (\beta = 1 \text{ を使用}) \quad (3)$$

ここでは，検出された発話区間と正解の発声区間の一部分が重なり合い，かつ，その区間の単語も正解単語と一致した場合のみを「正しく認識した発話区間」と判定した．ただし，音声入力対象の単語に先行するつなぎ語の部分は，ユーザにとっては音声認識を開始するためのトリガにすぎず，必ずしも正しく認識される必要はない．そこで，別のつなぎ語（別の種類の母音の引き延ばし）と認識した場合でも，誤りとは数えなかった（たとえば，「えー」を「あー」と認識しても誤りとしなかった）．

4.3 実験結果

まず，4 つの発話区間検出手法の比較評価をする前に，故意に発声した有声休止が雑音環境下でどの程度頑健に検出できるかを調査した．文献 9) のリアルタイム有声休止検出手法による検出率を，前述の 7 種類（(a)〜(g)）の各実環境雑音下においてそれぞれ求めた結果を**図 8** に示す．各実環境雑音における 5 つの棒グラフが，5 種類（0，10，20，30，40 dB）の各 SNR での 10 人の発声に対する検出率の平均値を表す．そして，それに付随する高低線が 10 人の発声の中での最大値と最小値を表す．さらに，この 10 人の発声に対する検出率の平均値を，7 種類すべての実環境雑音で平均した値を**図 9** に示す．5 種類の各 SNR において，棒グラフが 7 種類の実環境雑音に対する検出率の平均値を表し，それに付随する高低線が 7 種類の実環境雑音の中での最大値と最小値を表す．

次に，4 つの発話区間検出手法による性能の比較結果を，**図 10** に示す．図 10 (a)〜(g) は，7 種類（(a)〜(g)）の各実環境雑音に対応し，5 種類（0，10，20，30，40 dB）の各 SNR における 4 つの棒グラフが各手法の 10 人の発声に対する検出性能（F 値）の平均値を表す．そして，それに付随する高低線が 10 人の発声の中での最大値と最小値を表す．一方，図 10 (h)

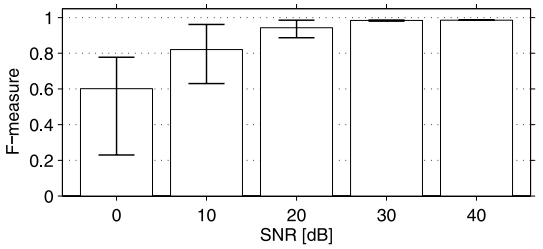


図 9 5 種類（0，10，20，30，40 dB）の各 SNR の実環境雑音下での有声休止検出率（10 人の発声に対する平均値）：各 SNR における棒グラフが 7 種類の実環境雑音に対する検出率の平均値を表し，それに付随する高低線が 7 種類の実環境雑音の中での最大値と最小値を表す

Fig. 9 Filled-pause detection rates with five different SNR conditions of noisy environments.

は，この 10 人の発声に対する検出性能の平均値を，7 種類すべての実環境雑音で平均した値を示す．5 種類の各 SNR において，棒グラフが 7 種類の実環境雑音に対する検出性能の平均値を表し，それに付随する高低線が 7 種類の実環境雑音の中での最大値と最小値を表す．

4.4 考　　察

まず音声スタータの有声休止検出率に関しては，図 8，図 9 によれば，SNR 30，40 dB ではほぼ完全に有声休止を検出でき，SNR 0，10 dB の高雑音環境下でも雑音によっては比較的高い性能で検出できていた．この結果から，我々の用いたリアルタイム有声休止検出手法が雑音に頑健であり，音声スタータの目的に使用できることが分かる．

次に音声スタータの発話区間検出性能に関しては，図 10 (h) の全体の結果によれば，音声スタータは他の手法で性能が出にくいような雑音の大きい環境である SNR 0〜20 dB において，他の手法に比べて性能が高く，有効であることが分かる．特に，SNR 10 dB で他の手法との差が大きく，効果的であった．

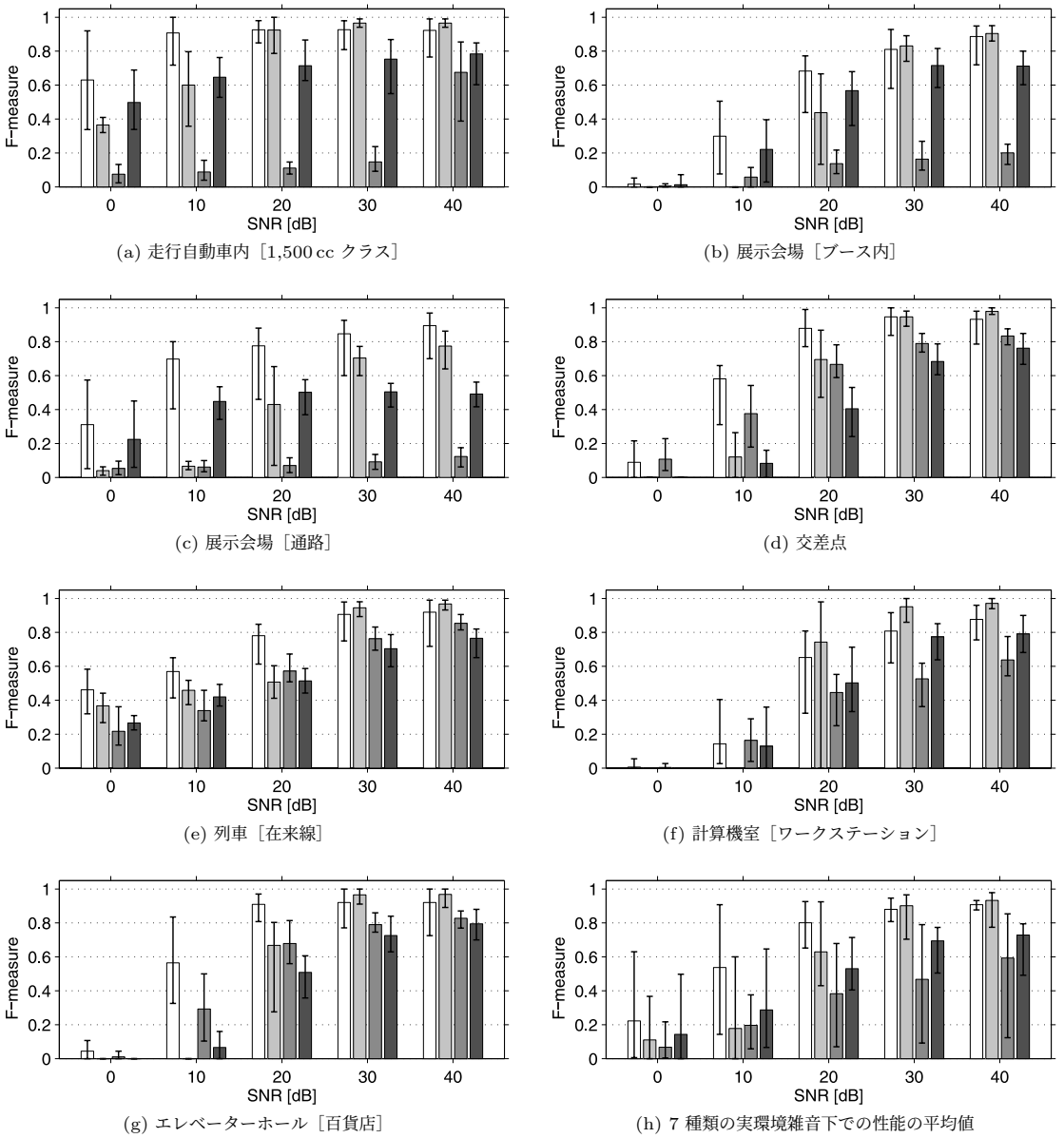
図 10 (a)〜(g) の個別の結果を考察すると，(a) の SNR 0，10 dB の結果，(b)，(d)，(g) の SNR 10，20 dB の結果，(c) の SNR 0〜40 dB の結果，(e) の SNR 0〜20 dB の結果において，音声スタータは他の手法よりも性能が高かった．特に，(a) の結果では音声スタータの性能の F 値自体が全体に高く，(c) の結果ではすべての SNR で他の手法よりも有効であった．図 8 から，(a) と (c) では有声休止検出の検出率自体も高かったことが分かる．(a) は低域にエネルギーが集中していて非定常な雑音を含む条件，(c) は人の話し声や音楽による雑音の多い条件であり，こうした雑音に対して音声スタータは頑健な傾向があることが分かる．

一方，(f) の結果では，音声スタータは他の手法と比べて性能が高くはなかった．これは，計算機のファンがうなり続けている定常雑音が，有声休止として誤検出されやすかったためである．実際に図 8 によれば，SNR 0，10 dB での有声休止の検出率自体が高くなかったことが分かる．また，(e) の結果では，ほかに比べて 4 つの発話区間検出手法の性能の差が小さい傾向にあった．これは，列車が通過しているときは激しい走行音がして音声認識が困難となり，ほとんどの手法が誤検出となってしまう一方，列車が通過していないときは雑音の音量が小さく，各手法にとって認識しやすい状況となっていたためである．

音声認識で用いられることの多い，零交差数と短時間エネルギーを用いる方法は，SNR 30，40 dB のような雑音の音量が小さい状況では有効であった（ただし，雑音 (c) の場合には人の話し声に起因する誤認識が多かったため，SNR 30，40 dB でも音声スタータの方が性能が高かった）．これは発話区間を誤検出する要因がほとんどなければ当然の結果であり，インタフェースの観点からも，誤検出さえなければ有声休止やキーワードを余分に発声しないですむ方が望ましい．雑音の音量が小さい状況のみであれば，従来どおりこの方法を用いればよい．一方，SNR 0〜20 dB のように雑音の音量が大きくなってくると，零交差数と短時間エネルギーを用いる方法よりは，他の方法の方が有効なことが多かった．

今回のキーワードを用いる方法の実装では，実質的には，ショートポーズセグメンテーションの認識対象の語句をキーワードの分だけ長くし，そのキーワード部分に棄却処理を加えたものとなっている．そのため，棄却処理が適切に働けばキーワードを用いる方法は有効だが，音声認識誤りや棄却の誤り等が発話区間の誤検出の要因となるため，雑音の音量が小さい場合には，より単純な零交差数と短時間エネルギーを用いる方法の方が性能が高いことが多かった．なお，音声スタータはキーワードを用いる方法よりもつねに性能が高かったが，これは，図 8，図 9 に示したように有声休止は雑音環境下でも頑健に検出できるためである．上述したようにインタフェースの観点からは余分な発声のない方が望ましいが，雑音環境下では，そうした余分な発声をしたとしても発話区間検出性能を上げることが重要となる．その意味でも，余分な発声をするのであれば性能が高い方が好ましく，音声スタータの方がキーワードを用いる方法よりも優れていると考えられる．

本実験では，音声スタータ用に「えー」と「あのー」



(a) 走行自動車内［1,500 cc クラス］
(b) 展示会場［ブース内］
(c) 展示会場［通路］
(d) 交差点
(e) 列車［在来線］
(f) 計算機室［ワークステーション］
(g) エレベーターホール［百貨店］
(h) 7 種類の実環境雑音下での性能の平均値

図 10 7 種類（(a)〜(g)）の各実環境雑音下での 4 つの発話区間検出手法の F 値（音声認識性能も考慮した発話区間検出性能）の比較結果：(a)〜(g) は，7 種類の各実環境雑音下において 4 つの手法の検出性能を比較した結果で，5 種類の各 SNR における 4 つの棒グラフが各手法の 10 人の発声に対する検出性能の平均値を表し，それに付随する高低線が 10 人の発声の中での最大値と最小値を表す．一方 (h) は，7 種類の実環境雑音に対する検出性能の平均値を比較した結果で，5 種類の各 SNR における 4 つの棒グラフが各手法の 7 種類の実環境雑音に対する検出性能の平均値を表し，それに付随する高低線が 7 種類の実環境雑音の中での最大値と最小値を表す

Fig. 10 F-measure comparison of the four endpoint detection methods in seven different noisy environments.

の 2 種類を用意し，キーワードを用いる方法用に「もしもし」と「コンピュータ」の 2 種類を用意して評価データを作成した．図 10 等には図示していないが，実験の結果，「えー」の場合と「あのー」の場合の発話区間検出性能は同等であり（有声休止検出率も同等であった），「もしもし」の場合と「コンピュータ」の場合の性能も同等であった．音声スタータのようにユーザが発話開始時につねに有声休止を発声することをルール化するアプローチの場合，事前に特定の有声休止 1 つに絞って教示することも可能だが，本研究のようにあえて絞らずに任意の母音の引き延ばしを許しても，性能上は問題ないと考えられる．

5. おわりに

本論文では，音声の非言語情報である有声休止を積極的に利用し，ユーザが音声だけで発話区間の始端を指定できる新たな音声入力インタフェース機能「音声スタータ」を提案した．音声スタータは，特別な訓練をせずに利用でき，マイク以外のデバイスが不要なだけでなく，雑音環境下で頑健に発話区間を検出できるという利点も持つ．実際に，従来の発話区間検出手法と比較評価した結果，高雑音環境下において有効であることを確認した．

音声スタータは，音声インタフェースでの非言語情報の新しい活用法を切り開くことで，音声の持つ潜在能力を引き出すことを目指した一連の「音声補完シリーズ」研究の第 3 弾に位置づけられる．第 1 弾の「音声補完」[12),20)] では有声休止によって補完機能を呼び出し，第 2 弾の「音声シフト」[21)] では声の高さによって入力モードを切り替える機能を実現した．それに対して音声スタータでは，雑音環境下の音声認識にインタフェースの観点から取り組み，有声休止を発話開始のトリガとして活用することで，通常だったら他のデバイスが必要な機能をマイク入力だけで実現した．今後は，音声スタータを用いた応用システムを構築していくとともに，他の非言語情報を活用した新たな音声インタフェースの可能性についても探求していく予定である．

参考文献

1) Rabiner, L.R. and Sambur, M.R.: An algorithm for determining the endpoints of isolated utterances, *The Bell System Technical J.*, Vol.54, No.2, pp.297–315 (1975).
2) Murai, K., Kumatani, K. and Nakamura, S.: A robust end point detection by speaker's facial motion, *International Workshop on Hands-Free Speech Communication* (*HSC 2001*), pp.199–202 (2001).
3) 松坂要佐，東條剛史，小林哲則：グループ会話に参与する対話ロボットの構築，信学論（D-II），Vol.J84-D-II, No.6, pp.898–908 (2001).
4) Bou-Ghazale, S.E. and Assaleh, K.: A robust endpoint detection of speech for noisy environments with application to automatic speech recognition, *Proc. ICASSP 2002*, pp.3808–3811 (2002).
5) Martin, A., Charlet, D. and Mauuary, L.: Robust speech/non-speech detection using LDA applied to MFCC, *Proc. ICASSP 2001*, pp.237–240 (2001).
6) Huang, L.-S. and Yang, C.-H.: A novel approach to robust speech endpoint detection in car environments, *Proc. ICASSP 2000*, pp.1751–1754 (2000).
7) Segawa, O., Takeda, K. and Itakura, F.: Continuous speech recognition without endpoint detection, *Proc. ICASSP 2001*, pp.245–248 (2001).
8) 河原達也，加藤一臣，南篠浩輝，李　晃伸：話し言葉音声認識のための言語モデルとデコーダの改善，情報処理学会研究報告音声言語情報処理 2001-SLP-36-3, pp.15–22 (2001).
9) 後藤真孝，伊藤克亘，速水　悟：自然発話中の有声休止箇所のリアルタイム検出システム，信学論（D-II），Vol.J83-D-II, No.11, pp.2330–2340 (2000).
10) 内藤正樹，黒岩眞吾，山本誠一，武田一哉：部分文仮説のゆう度を用いた連続音声認識のための音声区間検出法，信学論（D-II），Vol.J80-D-II, No.11, pp.2895–2903 (1997).
11) 井ノ上直己, 中村　誠, 酒寄信一, 山本誠一, 谷戸文廣：単語固有セルでのゆう度判定を用いた音声認識処理の高速化手法，信学論（D-II），Vol.J79-D-II, No.12, pp.2110–2116 (1996).
12) 後藤真孝，伊藤克亘，秋葉友良，速水　悟：音声補完：音声入力インタフェースへの新しいモダリティの導入，コンピュータソフトウェア（日本ソフトウェア科学会論文誌），Vol.19, No.4, pp.10–21 (2002).
13) 鹿野清宏，伊藤克亘，河原達也，武田一哉，山本幹雄：IT Text 音声認識システム，オーム社 (2001).
14) Lee, A., Kawahara, T. and Shikano, K.: Julius — An open source real-time large vocabulary recognition engine, *Proc. Eurospeech 2001*, pp.1691–1694 (2001).
15) Hayamizu, S., Itou, K. and Tanaka, K.: Detection of unknown words in large vocabulary speech recognition, *J. Acoust. Soc. Jpn.* (*E*), Vol.16, No.3, pp.165–171 (1995).

16) 村上仁一，嵯峨山茂樹：自由発話音声認識における音響的および言語的な問題点の検討，信学技報 SP91-100，pp.71–78 (1991).
17) 板橋秀一：騒音データベースと日本語共通音声データ DAT 版，日本音響学会誌，Vol.47, No.12, pp.951–953 (1991).
18) Itou, K., Yamamoto, M., Takeda, K., Takezawa, T., Matsuoka, T., Kobayashi, T., Shikano, K. and Itahashi, S.: The design of the newspaper-based Japanese large vocabulary continuous speech recognition corpus, *Proc. ICSLP 98*, pp.3261–3264 (1998).
19) van Rijsbergen, C.J.: *Information retrieval*, 2nd edition, Butterworths (1979).
20) 後藤真孝：解説 “音声補完：言い淀むと助けてくれる音声インタフェース”，情報処理（情報処理学会誌），Vol.43, No.11, pp.1210–1216 (2002).
21) 尾本幸宏，後藤真孝，伊藤克亘，小林哲則：音声シフト：音高の意図的な変化を利用した音声入力インタフェース，信学論（D-II），Vol.J88-D-II, No.3, pp.469–479 (2005).

(平成 18 年 6 月 21 日受付)
(平成 19 年 2 月 1 日採録)

後藤　真孝（正会員）

1993 年早稲田大学理工学部電子通信学科卒業．1998 年同大学大学院博士後期課程修了．同年電子技術総合研究所（2001 年に産業技術総合研究所に改組）に入所し，現在に至る．2000 年から 2003 年まで科学技術振興事業団さきがけ研究 21「情報と知」領域研究員，2005 年から筑波大学大学院助教授（連携大学院）を兼任．博士（工学）．音楽情報処理，音声言語情報処理等に興味を持つ．2000 年 WISS2000 論文賞・発表賞，2001 年日本音響学会粟屋潔学術奨励賞・ポスター賞，2003 年インタラクション 2003 ベストペーパー賞，2005 年情報処理学会論文賞等 19 件受賞．電子情報通信学会，日本音響学会，日本音楽知覚認知学会各会員．

北山　広治

2002 年早稲田大学理工学部電気電子情報工学科卒業．2004 年同大学大学院修士課程修了．同年（株）東芝入社．大学では音声インタフェースの研究に従事．東芝では動画コーデック LSI の研究開発を担当．

伊藤　克亘（正会員）

博士（工学）．1993 年電子技術総合研究所入所．2003 年名古屋大学大学院情報科学研究科助教授．2006 年法政大学情報科学部教授．現在に至る．音声を主とした自然言語全般に興味を持つ．

小林　哲則（正会員）

1985 年早稲田大学大学院博士課程修了．工学博士．同年法政大学工学部電気工学科講師．同助教授を経て，1991 年早稲田大学理工学部電気工学科助教授．1997 年電気電子情報工学科教授．現在，コンピュータ・ネットワーク工学科教授．MIT，ATR，NHK 技研等の客員研究員を歴任．音声情報処理，動画像処理等知覚情報システムの基礎研究およびその応用としての会話ロボットの研究に従事．2001 年度電子情報通信学会論文賞受賞．電子情報通信学会，日本音響学会，言語処理学会等の会員．